

⑯ BUNDESREPUBLIK

DEUTSCHLAND



DEUTSCHES

PATENT- UND
MARKENAMT

Offenlegungsschrift

DE 100 30 991 A 1

3 US 6,848,064

⑯ Int. Cl. 7:

G 06 F 11/30

G 05 B 23/02

⑯ Aktenzeichen: 100 30 991.7
⑯ Anmeldetag: 30. 6. 2000
⑯ Offenlegungstag: 10. 1. 2002

⑯ Anmelder:

Robert Bosch GmbH, 70469 Stuttgart, DE

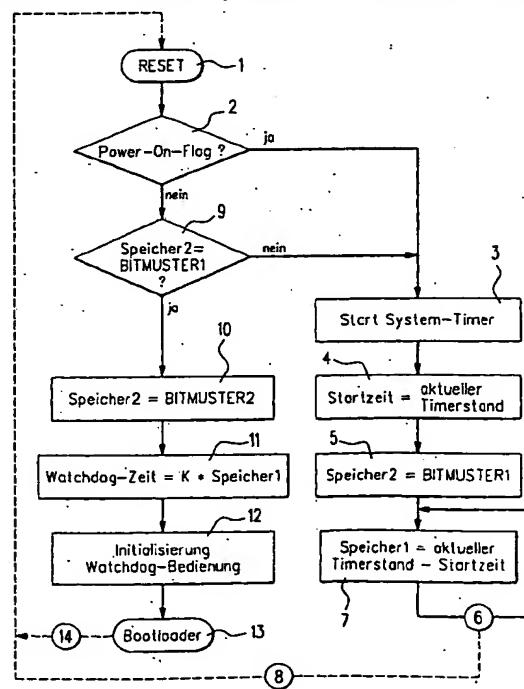
⑯ Erfinder:

Bolz, Peter, 71706 Markgröningen, DE

Die folgenden Angaben sind den vom Anmelder eingereichten Unterlagen entnommen

⑯ Verfahren und Vorrichtung zur Synchronisation eines Rechners und einer Überwachungsschaltung

⑯ Die Erfindung betrifft ein Verfahren zum Betreiben mindestens eines Rechners und mindestens einer dem mindestens einen Rechner zugeordneten und nach einem bestimmten zeitlichen Muster von dem Rechner zu bedienenden Überwachungsschaltung (Watchdog), insbesondere für eine Steuer- oder Sicherheitseinrichtung für ein Kraftfahrzeug, wobei der Rechner rückgesetzt wird, falls die Überwachungsschaltung nicht nach dem zeitlichen Muster bedient wird. Um auf möglichst einfache Weise eine Synchronisation des Rechners auf die Überwachungsschaltung zu ermöglichen, wird vorgeschlagen, dass nach einem Hochfahren (1) des Rechners aus einem Rücksetzmodus in einen Arbeitsmodus eine Zeitbasis ermittelt wird, die nach dem Hochfahren des Rechners bis zu einem Rücksetzen des Rechners auf Grund einer fehlenden Bedienung der Überwachungsschaltung nach dem zeitlichen Muster vergeht, und dass die Überwachungsschaltung in zeitlichen Abständen bedient wird, die in Abhängigkeit von der Zeitbasis ermittelt werden (11).



DE 100 30 991 A 1

BEST AVAILABLE COPY

BUNDESDRUCKEREI 11.01 101 620/243/1

14

Beschreibung

[0001] Die vorliegende Erfindung betrifft ein Verfahren zum Betreiben mindestens eines Rechners und mindestens einer dem mindestens einem Rechner zugeordneten Überwachungsschaltung (Watchdog). Der Rechner dient insbesondere für eine Steuer- oder Sicherheitseinrichtung für ein Kraftfahrzeug. Die Überwachungsschaltung muss nach einem bestimmten zeitlichen Muster von dem Rechner bedient werden. Falls die Überwachungsschaltung nicht nach dem zeitlichen Muster bedient wird, wird der Rechner rückgesetzt.

[0002] Die Erfindung betrifft außerdem ein Steuergerät mit mindestens einem Rechner und mindestens einer dem mindestens einem Rechner zugeordneten und nach einem bestimmten zeitlichen Muster von dem Rechner zu bedienenden Überwachungsschaltung (Watchdog), insbesondere für eine Sicherheitseinrichtung für ein Kraftfahrzeug.

[0003] Schließlich betrifft die vorliegende Erfindung ein Steurelement, insbesondere ein Read-Only-Memory oder ein Flash-Memory, für ein derartiges Steuergerät.

Stand der Technik

[0004] In vielen Steuergeräten und in zunehmendem Maße auch in Kraftfahrzeugen kommen als Mikrocontroller bzw. Mikroprozessoren ausgebildete Rechner zur Steuerung bzw. Regelung beliebiger Funktionen zum Einsatz. Die zu steuernden/regelnden Funktionen in einem Kraftfahrzeug sind bspw. die Brennkraftmaschine, eine Klimaanlage oder sicherheitskritische Funktionen wie eine elektronische Lenkung (steer-by-wire), ein elektronisches Gaspedal, ein Antiblockierbremsystem (ABS) oder ein Passagier-Rückhaltesystem (Airbag, Gurtstraffer, rückpralldämpfende Kopfstütze, etc.).

[0005] Mikroprozessoren beinhalten nur ein Rechenwerk, während Mikrocontroller auch Speicher, bspw. Random-Access-Memory (RAM), Read-Only-Memory (ROM), Electrically-Erasable-Programmable-Read-Only-Memory (EEPROM) oder Flash-Memory, und Peripheriefunktionen, bspw. einen Digital/Analog-Wandler oder Analog/Digital-Wandler, umfassen. Mikrocontroller können Fehlfunktionen zeigen, die durch äußere Einflüsse, wie bspw. elektromagnetische Einstrahlung oder statische Entladung, verursacht sein können. Eine weitere Ursache für Fehlfunktionen können Störungen auf Grund von Software-Fehlern in einem bestimmten Betriebsmodus sein.

[0006] Um solche Fehlfunktionen eines Mikrocontrollers erkennen zu können, besitzen moderne Steuergeräte mehr oder weniger aufwendige Hardware-Überwachungsschaltungen, sog. Watchdogs. Ein Watchdog ist bspw. auf einem anwendungsspezifischen integrierten Schaltkreis (sog. Application Specific Integrated Circuit; ASIC) realisiert. Ein Watchdog muss von dem Mikrocontroller nach einem bestimmten zeitlichen Muster, bspw. in festen zeitlichen Abständen, bedient bzw. angesteuert werden. Unterbleibt eine Bedienung des Watchdogs geht dieser von einer Fehlfunktion des Mikrocontrollers aus und veranlasst ein Rücksetzen des Mikrocontrollers.

[0007] Anschließend wird der Mikrocontroller aus dem Rücksetzmodus wieder in einen Arbeitsmodus hochgefahren. Das Rücksetzen des Mikrocontrollers umfasst ein Rücksetzen sämtlicher Peripheriefunktionen und eine Initialisierung des Mikroprozessors. Beim Hochfahren des Mikrocontrollers aus dem Rücksetzmodus werden Funktions- tests durchgeführt, um eine Fehlfunktion des Mikrocontrollers detektieren und möglichst auch diagnostizieren zu können. Im Gegensatz zum Rücksetzen und Hochfahren des

Mikrocontrollers nach einem Aus- und Einschalten des Mikrocontrollers liegt beim Rücksetzen und Hochfahren des Mikrocontrollers nach einem Rücksetzbefehl des Watchdog die Versorgungsspannung ohne Unterbrechung an dem Mikrocontroller an, so dass Dateninhalte der Speicher des Mikrocontrollers erhalten bleiben.

[0008] Das Bedienen des Watchdogs durch den Mikrocontroller kann auf unterschiedliche Weise erfolgen, bspw. durch ein einfaches digitales Signal über eine elektrische Verbindung. Der Watchdog kann aber auch durch eine zeitliche Sequenz von Signalen über mehrere elektrische Verbindungen bedient werden. Die Signale müssen nach einem bestimmten zeitlichen Muster von dem Mikrocontroller an den Watchdog übermittelt werden. Ein einfacher Watchdog muss bspw. in regelmäßigen zeitlichen Abständen vor Ablauf einer bestimmten Zeitspanne bedient werden. Ein Fenster-Watchdog muss innerhalb eines festen Zeitfensters regelmäßig bedient werden. Ein Fenster-Watchdog veranlasst ein Rücksetzen des Mikrocontrollers sowohl wenn der Watchdog zu früh (vor Beginn des Zeitfensters) als auch zu spät (nach dem Ende des Zeitfensters) bedient wird.

[0009] Aus der DE 40 23 700 A1 ist eine Watchdog-Überwachungsschaltung bekannt, die in regelmäßigen zeitlichen Abständen von einem Mikrocontroller durch ein Watchdog-Signal bedient wird. Liegt die Frequenz der aus den einzelnen Watchdog-Signalen entstehenden Signalfolge außerhalb eines bestimmten gültigen Frequenzfensters, detektiert der Watchdog eine Fehlfunktion des Mikrocontrollers.

[0010] Ein Watchdog besitzt von dem Mikrocontroller unabhängige Einrichtungen zum Erzeugen von Watchdog-Zeiten. Ein Watchdog kann unabhängig von einem Mikrocontroller auf ein festes zeitliches Muster, bspw. einen festen zeitlichen Abstand bis zu einer erneuten Bedienung, eingestellt werden. Bei der schärferen Anforderung eines Fenster-Watchdog kann unabhängig von einem Mikrocontroller der Beginn und das Ende des Zeitfensters fest eingestellt werden. Das zeitliche Muster nach dem der Watchdog bedient werden muss, kann also variieren ohne dass dies von außen anhand des Watchdogs oder des eingesetzten Mikrocontrollers erkennbar wäre.

[0011] Mikrocontroller werden in zunehmendem Maße mit frei programmierbaren Programmspeicher gefertigt, z. B. in der Flash-Technik. Durch die Flash-Technik ist es möglich, dass eine Mikrocontroller-Bauart in verschiedenen Applikationen mit applikationsspezifischer Software programmiert wird. Die Programmierung kann bereits beim Halbleiterhersteller oder erst vor Ort beim Anwender erfolgen. Zur Programmierung kann der Mikrocontroller in ein separates Programmiergerät eingelegt werden.

[0012] In zunehmendem Maße erfolgt die Programmierung jedoch beim Anwender bei in einer Zielhardware – bspw. dem Steuergerät einer Sicherheitseinrichtung für ein Kraftfahrzeug – eingelegten Mikrocontroller. Die Programmierung in der Zielhardware wird in der Regel durch serielle Kommunikation zwischen einem externen Rechner und einem in dem Mikrocontroller ablaufähigen Hilfsprogramm gesteuert. Das Hilfsprogramm wird auch als Bootloader bezeichnet und beinhaltet die Programmieralgorithmen und das Datenhandling. Die Programmierung kann z. B. in der Fertigungslinie des Steuergeräteherstellers oder aber unmittelbar in einem Kraftfahrzeug erfolgen. Die serielle Kommunikation erfolgt in einem Kraftfahrzeug üblicherweise über eine sog. K-Leitung oder über einen Controller-Area-Network (CAN)-Bus.

[0013] Unterschiedliche Applikationen des Mikrocontrollers bringen es mit sich, dass sowohl eine Mikrocontroller-Bauart mit unterschiedlichen Prozessor-Taktraten betrieben wird als auch die Watchdogs mit unterschiedlichen zeitli-

chen Mustern zu bedienen sind. Dies stellt insbesondere während der Programmierung des Mikrocontrollers mittels des Bootloaders, ein großes Problem dar, da dem Microcontroller vor der Programmierung nicht bekannt sein kann, nach welchem zeitlichen Muster der Watchdog zu bedienen ist.

[0014] Um eine Bedienung des Watchdogs sicherzustellen, muss nach dem Stand der Technik für eine bestimmte Prozessor-Taktfrequenz und für ein bestimmtes zeitliches Muster zur Bedienung des Watchdogs jeweils eine individuelle Bootloader-Software vorhanden sein. Dies ist jedoch aufwendig, speicherplatzintensiv und teuer.

[0015] Aus dem Stand der Technik sind des weiteren Bootloader bekannt, die verschiedene feste zeitliche Muster zum Bedienen des Watchdogs vorschreiben. Das, zeitliche Muster ist z. B. über Mikrocontroller-Ports oder interne nicht flüchtige Speicherzellen konfigurierbar, so dass sich eine geringe Flexibilität der Bootloader ergibt. Neben der stark eingeschränkten Flexibilität dieser bekannten Bootloader ist ein weiterer Nachteil dieser bekannten Bootloader der relativ große Hardware- und Softwareaufwand.

[0016] Der vorliegenden Erfindung liegt die Aufgabe zu Grunde, auf eine möglichst einfache Weise eine Synchronisation eines Rechners mit einer dem Rechner zugeordneten und nach einem bestimmten zeitlichen Muster von dem Rechner zu bedienenden Überwachungsschaltung (Watchdog) zu erzielen.

[0017] Zur Lösung der Aufgabe wird ausgehend von dem Verfahren der eingangs genannten Art vorgeschlagen, dass nach einem Hochfahren des Rechners aus einem Rücksetzmodus in einen Arbeitsmodus eine Zeitbasis ermittelt wird, die nach dem Hochfahren des Rechners bis zu einem Rücksetzen des Rechners auf Grund einer fehlenden Bedienung der Überwachungsschaltung nach dem zeitlichen Muster vergeht, und dass die Überwachungsschaltung in zeitlichen Abständen bedient wird, die in Abhängigkeit von der Zeitbasis ermittelt werden.

Vorteile der Erfindung

[0018] Erfindungsgemäß wird das zeitliche Muster, mit dem der Watchdog bedient werden muss, um zu verhindern, dass der Watchdog ein Rücksetzen des Rechners veranlasst, für die jeweilige Applikation des Rechners individuell ermittelt. Die verschiedenen Applikationen können sich in der Taktrate des Mikroprozessors oder in dem zeitlichen Muster des Watchdogs unterscheiden. Mit dem erfindungsgemäßen Verfahren kann der Rechner auf eine besonders einfache Weise auf die Überwachungsschaltung synchronisiert werden.

[0019] Zur Ermittlung des zeitlichen Musters wird der Rechner zunächst in an sich bekannter Weise hochgefahren. Nach diversen Initialisierungsvorgängen unmittelbar nach dem Hochfahren muss die Überwachungsschaltung bedient werden. Gemäß der vorliegenden Erfindung unterbleibt jedoch die Bedienung der Überwachungsschaltung und es wird gewartet, bis die Überwachungsschaltung ein Rücksetzen des Rechners wegen fehlender Bedienung nach dem vorgegebenen zeitlichen Muster veranlasst. Die Zeit, die vom Hochfahren des Rechners bis zum Rücksetzen vergeht, wird als Zeitbasis erfasst. In Abhängigkeit dieser Zeitbasis werden zeitliche Abstände ermittelt, die zwischen einer Bedienung der Überwachungsschaltung und der nächsten Bedienung vergehen.

[0020] Gemäß einer vorteilhaften Weiterbildung der vorliegenden Erfindung wird vorgeschlagen, dass das Verfahren ausgeführt wird bevor mit Hilfe eines in dem Rechner gespeicherten Hilfsprogramms (Bootloader) ein Applikati-

onsprogramm in einen Programmspeicher des Rechners geladen wird. Die im Rahmen des erfindungsgemäßen Verfahrens ermittelten zeitlichen Abstände, in denen die Überwachungsschaltung von dem Rechner bedient werden muss, werden also bei einer Programmierung des Rechners mit einem Applikationsprogramm berücksichtigt. Dadurch kann ein einziger Bootloader unabhängig von der Taktrate eines Mikroprozessors des Rechners und unabhängig von dem zeitlichen Muster zum Bedienen der Überwachungsschaltung arbeiten. Das zeitliche Muster zum Böden der Überwachungsschaltung wird individuell vor jeder Programmierung des Rechners ermittelt. Die Überwachungsschaltung wird in den ermittelten zeitlichen Abständen bedient, so dass ein Rücksetzen des Rechners durch die Überwachungsschaltung während der Programmierung des Rechners verhindert wird.

[0021] Gemäß einer bevorzugten Ausführungsform der vorliegenden Erfindung wird vorgeschlagen, dass die zeitlichen Abstände durch Multiplikation der Zeitbasis mit einem Korrekturfaktor ermittelt werden. Der Korrekturfaktor wird vorteilhaftweise kleiner als 1 – vorzugsweise zwischen 0,75 und 0,9 – gewählt. Die Zeitbasis gibt diejenige Zeitspanne wieder, nach der der Watchdog spätestens hätte bedient werden müssen, um zu verhindern, dass die Überwachungsschaltung ein Rücksetzen des Rechners veranlasst. Durch den Korrekturfaktor werden die zeitlichen Abstände etwas verringert, so dass der Watchdog etwas vor dem spätest möglichen Zeitpunkt bedient wird. Auf diese Weise kann verhindert werden, dass auf Grund einer geringfügigen zeitlichen Verschiebung der Bedienung des Watchdogs, bspw. auf Grund von Fertigungstoleranzen des Mikrocontrollers oder vor, verlängerten Signallaufzeiten, zu späteren Zeitpunkten hin durch die Überwachungsschaltung fälschlicherweise ein Rücksetzen des Rechners veranlasst wird.

[0022] Gemäß einer anderen vorteilhaften Weiterbildung der vorliegenden Erfindung wird vorgeschlagen, zum Ermitteln der Zeitbasis nach dem Hochfahren des Rechners aus einem Rücksetzmodus in einen Arbeitsmodus ein Zähler hochgezählt und der Zählerstand in einem ersten Speicherlement, vorzugsweise einem Random-Access-Memory (RAM), abgelegt wird. Der Zähler speichert die Differenz des Zählerstandes eines System-Timers nach dem Hochfahren des Rechners (Systemstart) zu dem Zählerstand während des Wartens auf ein Rücksetzen durch den Watchdog. Bei manchen Mikrocontrollern wird der System-Timer automatisch nach dem Hochfahren des Rechners gestartet. Bei anderen Mikrocontrollern muss der System-Timer nach dem Hochfahren des Rechners durch einen besonderen Befehl gestartet werden. Der Zählerstand wird in einem Speicherlement abgelegt, an dem während des Rücksetzens und des erneuten Hochfahrens des Rechners ununterbrochen eine Versorgungsspannung anliegt, so dass der Inhalt des Speicherlements auch nach einem erneuten Hochfahren des Rechners nicht verloren geht.

[0023] Gemäß einer weiteren bevorzugten Ausführungsform der vorliegenden Erfindung wird vorgeschlagen, dass der Zählerstand nach einem erneuten Hochfahren des Rechners nach dem Rücksetzen des Rechners aufgrund einer fehlenden Bedienung der Überwachungsschaltung der Zählerstand aus dem ersten Speicherlement ausgelesen und als Zeitbasis zum Bedienen der Überwachungsschaltung herangezogen wird. Der Zählerstand entspricht demjenigen Zeitpunkt, zu dem der Watchdog nach dem Hochfahren des Rechners spätestens hätte bedient werden sollen, um ein durch die Überwachungsschaltung veranlasstes Rücksetzen des Rechners zu verhindern. Auf Grundlage des Zählerstandes kann eine Zeitbasis und können zeitliche Abstände ermittelt werden, die zum Bedienen des Watchdogs herange-

zogen wird.

[0024] Vorteilhafterweise wird nach einem Hochfahren des Rechners aus einem Rücksetzmodus in einen Arbeitsmodus ein zweites Speicherelement mit einem ersten vorgebbaren Bitmuster belegt und der Inhalt des zweiten Speicherelements zum Erkennen eines erneuten Hochfahrens des Rechners bei jedem Hochfahren des Rechners abgefragt. Entspricht der Inhalt des zweiten Speicherelements beim Hochfahren des Rechners dem ersten Bitmuster, handelt es sich um ein erneutes Hochfahren des Rechners und nicht um ein Hochfahren nach einem Aus- und Einschalten (Power-On) des Rechners. Bei einem erneuten Hochfahren des Rechners wird dann der Inhalt des ersten Speicherelements ausgelesen und es werden die zeitlichen Abstände zum Bedienen der Überwachungsschaltung bestimmt.

[0025] Gemäß einer weiteren bevorzugten Ausführungsform der vorliegenden Erfindung wird vorgeschlagen, dass nach einem erneuten Hochfahren des Rechners das zweite Speicherelement mit einem von dem ersten Bitmuster unterschiedlichen zweiten vorgebbaren Bitmuster belegt wird. Es ist denkbar, dass es während der Ausführung des erfindungsgemäßen Verfahrens tatsächlich zu einer Fehlfunktion des Rechners und infolgedessen zu einem Rücksetzen des Rechners kommt. In einem solchen Fall hat der in dem ersten Speicherelement abgelegte Zählerstand noch nicht seinen Endwert erreicht. Der Zählerstand gibt nämlich nicht den spätest möglichen Zeitpunkt zum Bedienen der Überwachungsschaltung, sondern lediglich den Zeitpunkt wieder, an dem der Rechner auf Grund der Fehlfunktion rückgesetzt wurde. Durch die vorgeschlagene Ausführungsform kann verhindert werden, dass ein in dem ersten Speicherelement abgelegter falscher Zählerstand als Zeitbasis zum Bedienen der Überwachungsschaltung herangezogen wird. Die Überwachungsschaltung wird zunächst mit dem ermittelten falschen zeitlichen Abstand bedient und veranlasst infolgedessen ein Rücksetzen des Rechners. Nach dem erneuten Hochfahrendes Rechners wird das erfindungsgemäße Verfahren dann noch einmal durchlaufen und dieses Mal der richtige zeitliche Abstand ermittelt, sofern nicht wieder ein Rücksetzen des Rechners auf Grund einer Fehlfunktion auftritt.

[0026] Zum Erkennen eines erneuten Hochfahrens des Rechners wird alternativ oder zusätzlich vorgeschlagen, dass bei jedem Hochfahren des Rechners überprüft wird, ob eine Markierung (Power-On-Flag) des Rechners zum Erkennen eines Hochfahrens des Rechners aus einem Rücksetzmodus in einen Arbeitsmodus nach einem Einschalten des Rechners gesetzt ist. Die Power-On-Flag ist ein Hardware-Bit, das nach einem erstmaligen Hochfahren des Rechners nach dem Einschalten gesetzt ist, und das nach dem Hochfahren des Rechners im Anschluss an ein Rücksetzen jedoch nicht gesetzt ist. Durch Abfragen der Power-On-Flag kann festgestellt werden, ob der Rechner das erste Mal nach dem Einschalten des Rechners oder nach einem Rücksetzen des Rechners hochfährt. Eine Power-On-Flag ist allerdings nicht bei jedem Rechner verfügbar.

[0027] Von besonderer Bedeutung ist die Realisierung des erfindungsgemäßen Verfahrens in Form eines Steuerelements, das für ein Steuergerät insbesondere einer Sicherheitseinrichtung für ein Kraftfahrzeug vorgesehen ist. Dabei ist auf dem Steuerelement ein Programm abgespeichert, das auf einem Rechner, insbesondere auf einem Mikroprozessor, ablauffähig und zur Ausführung des erfindungsgemäßen Verfahrens geeignet ist. In diesem Fall wird also die Erfindung durch ein auf dem Steuerelement abgespeichertes Programm realisiert, so dass dieses mit dem Programm versehene Steuerelement in gleicher Weise die Erfindung darstellt wie das Verfahren, zu dessen Ausführung das Programm geeignet ist. Als Steuerelement kann insbesondere ein elektri-

sches Speichermedium zur Anwendung kommen, bspw. ein Read-Only-Memory oder ein Flash-Memory.

[0028] Als eine weitere Lösung der Aufgabe der vorliegenden Erfindung wird ausgehend von dem Steuergerät der eingangs genannten Art vorgeschlagen, dass das Steuergerät Mittel zum Ermitteln einer Zeitbasis aufweist, die nach einem Hochfahren des Rechners aus einem Rücksetzmodus in einen Arbeitsmodus bis zu einem Rücksetzen des Rechners auf Grund einer fehlenden Bedienung der Überwachungsschaltung nach dem zeitlichen Muster vergeht, und dass das Steuergerät zweite Mittel zum Bedienen der Überwachungsschaltung in der Zeitbasis entsprechenden zeitlichen Abständen aufweist.

[0029] Weitere Merkmale, Anwendungsmöglichkeiten 15 und Vorteile der Erfindung ergeben sich aus der nachfolgenden Beschreibung von Ausführungsbeispielen der Erfindung, die in der Zeichnung dargestellt sind. Dabei bilden alle beschriebenen oder dargestellten Merkmale für sich oder in beliebiger Kombination den Gegenstand der Erfindung, unabhängig von ihrer Zusammenfassung in den Patentansprüchen oder deren Rückbeziehung sowie unabhängig von ihrer Formulierung bzw. Darstellung in der Beschreibung bzw. in der Zeichnung. Es zeigt:

[0030] Fig. 2 ein Ablaufdiagramm eines erfindungsgemäßen Verfahrens gemäß einer bevorzugten Ausführungsform.
[0031] Fig. 1 zeigt eine Anordnung mit einem Watchdog und einem Microcontroller, welche erfindungsgemäß betrieben bzw. ausgestaltet ist.

30 Beschreibung der Ausführungsbeispiele

[0032] Fig. 1 zeigt einen Microcontroller μ C mit wenigstens einem Steuerelement SE, insbesondere wenigstens einem Speicherelement, z. B. ein Read-Only-Memory, Random-Access-Memory oder Flash-Memory, und einen Watchdog WD, welche miteinander verbunden sind.

[0033] Über den Port WDS des Mikrocontrollers kann das Watchdogsignal, also ein Signal, welches nach einem bestimmten zeitlichen Muster übertragen wird, an den Watchdog WD auf dessen WDS Port übertragen werden.

[0034] Mit einem Watchdog Port IRS, welcher mit einem Reset-Port R des Microcontrollers verbunden ist kann ein Reset des Mikrocontrollers durch den Watchdog ausgelöst werden. Dazu wird beispielsweise ein Impulsresetignal vom Watchdog und den Mikrocontroller μ C gesandt.

[0035] Dabei ist die Anordnung z. B. Teil eines Steuergerätes SG bzw. in einem Steuergerät SG untergebracht.

[0036] Der Watchdog und der Mikrocontroller werden dabei erfindungsgemäß synchronisiert, was anhand Fig. 2 näher erläutert wird.

[0037] Das in Fig. 2 dargestellte erfindungsgemäße Verfahren dient zum Synchronisieren eines Mikrocontrollers auf einen externen Watchdog. Ein Watchdog ist eine Hardware-Überwachungsschaltung, die von dem Mikrocontroller nach einem bestimmten zeitlichen Muster bedient werden muss. Wird der Watchdog nicht nach dem vorgegebenen zeitlichen Muster bedient, geht er von einer Fehlfunktion des Mikrocontrollers aus und veranlasst ein Rücksetzen des Mikrocontrollers. Das erfindungsgemäße Verfahren wird

60 insbesondere zu Beginn einer Programmierung des Mikrocontrollers ausgeführt. Zur Programmierung ist der Mikrocontroller in einer Zielhardware, bspw. in einem Steuergerät für eine Steuer- oder Sicherheitseinrichtung für ein Kraftfahrzeug, eingelegt. Die Programmierung in der Zielhardware wird in der Regel durch eine serielle Kommunikation zwischen einem externen Programmierrechner und einem auf dem Mikrocontroller ablauffähigen Hilfsprogramm gesteuert. Das Hilfsprogramm wird auch als Bootloader be-

zeichnet und beinhaltet die Programmieralgorithmen und das Datenhandling.

[0038] Als Bootloader im Sinne der vorliegenden Erfindung wird zum einen ein mikrocomputerspezifischer Bootloader bezeichnet, der in der Regel von dem Halbleiterhersteller in dem Mikrocontroller zur Verfügung gestellt wird oder von einem Anwender vor dem Einbau des Mikrocontrollers in der Zielhardware in einem separaten Programmiergerät einprogrammiert wird. Ein derartiger Bootloader ist üblicherweise einheitlich für viele verschiedene Applikationen und anwenderunabhängig. Zum anderen kann es sich bei dem Bootloader aber auch um eine von dem Anwender für eine bestimmte Applikation oder nach bestimmten Anforderungen des Anwenders (Kompatibilität mit anderen Bootloadern, einheitliche Bootloader für einen bestimmten Kraftfahrzeugtyp oder für einen Kraftfahrzeugherrsteller, etc.) geschaffene Software handeln.

[0039] Bei dem in Fig. 2 dargestellten Ausführungsbeispiel des erfindungsgemäßen Verfahrens erfolgt die Synchronisation des Mikrocontrollers auf den externen Watchdog durch eine Programmsequenz zu Beginn des Bootloaders. Zunächst wird der Watchdog von dem Mikrocontroller nicht bedient. Es wird darauf gewartet, dass, der Watchdog ein Rücksetzen des Mikrocontrollers veranlasst, da der Watchdog nicht nach dem vorgegebenen zeitlichen Muster bedient worden ist. Vom Hochfahren des Rechners bis zum Rücksetzen des Rechners auf Grund einer ausbleibenden Bedienung des Watchdogs wird eine Zeitbasis gemessen und aus der Zeitbasis dann eine Watchdog-Zeit bzw. ein zeitlicher Abstand zwischen zwei aufeinanderfolgenden Bedienungen des Watchdogs ermittelt. In diesem zeitlichen Abstand kann der Mikrocontroller dann den Watchdog während der Programmierung bedienen und so ein Rücksetzen des Mikrocontrollers verhindern.

[0040] Das erfindungsgemäße Verfahren beginnt bei Funktionsblock 1 mit einem Hochfahren des Rechners aus einem Rücksetzmodus in einen Arbeitsmodus. Der Rechner kann zum ersten Mal nach dem Einschalten des Rechners (sog. Kaltstart) oder nach einem Rücksetzbefehl des Watchdogs erneut hochfahren (sog. Warmstart).

[0041] In einem nachfolgenden Abfrageblock 2 wird überprüft, ob eine Power-On-Flag gesetzt ist. Die Power-On-Flag ist ein Hardware-Bit, das bei einem erstmaligen Hochfahren des Rechners nach dem Einschalten des Rechners gesetzt ist. Bei einem erneuten Hochfahren des Rechners auf Grund eines Rücksetzbefehls des Watchdogs ist die Power-On-Flag nicht gesetzt. Der Abfrageblock 2 ist nur bei solchen Mikrocontrollern vorhanden, die eine Power-On-Flag aufweisen. Die Abfrage der Power-On-Flag dient als zusätzliche Abfrage, ob der Rechner erstmalig nach dem Einschalten des Rechners oder nach einem Rücksetzbefehl des Watchdogs erneut hochfährt. Auf den Abfrageblock 2 könnte ohne Einschränkung der Funktionalität des erfindungsgemäßen Verfahrens auch verzichtet werden.

[0042] Falls der Rechner erstmalig nach dem Einschalten des Rechners hochfährt, wird zu Funktionsblock 3 verzweigt, wo ein Zähler gestartet wird. Der Zähler ist bspw. als ein System-Timer ausgebildet. Es gibt auch Mikrocontroller, die eine freilaufenden/selbststartenden System-Timer aufweisen (z. B. Motorola HC12), die nicht gestartet werden müssen. Bei solchen Mikrocontrollern kann in dem erfindungsgemäßen Verfahren auf den Funktionsblock 3 verzichtet werden.

[0043] Anschließend wird in einem Funktionsblock 4 der aktuelle Stand des System-Timers als Startzeit abgespeichert. In einem zweiten Speicherelement (Speicher2) wird in Funktionsblock 5 ein erstes vorgebares Bitmuster (BITMUSTER1) abgelegt. Anschließend geht das erfindungsge-

mäße Verfahren in eine Endlosschleife 6 über, in deren Verlauf in einem Funktionsblock 7 die Differenz des aktuellen Stands des System-Timers und der Startzeit in einem ersten Speicherelement (Speicher1) abgelegt wird. Das erste Speicherelement ist bspw. als ein Random-Access-Memory (RAM) ausgebildet, an dem während des Rücksetzens und des Hochfahrens des Rechners aufgrund eines Rücksetzbefehls des Watchdogs (Warmstart) ohne Unterbrechung eine Versorgungsspannung anliegt. Dadurch ist der Inhalt des ersten Speicherelements auch nach einem Rücksetzen des Rechners noch vorhanden, sofern die Versorgungsspannung stabil bleibt. Die Endlosschleife 6 wird so lange durchlaufen, bis der Watchdog ein Rücksetzen des Rechners veranlasst, da der Watchdog nicht nach dem vorgegebenen zeitlichen Muster bedient worden ist. Das Rücksetzen des Rechners ist in Fig. 2 durch eine gestrichelte Linie 8 dargestellt.

[0044] Nach dem Rücksetzen 8 des Rechners fährt der Rechner in Funktionsblock 1 erneut hoch. Da es sich um ein erneutes Hochfahren nach einem Rücksetzbefehl des Watchdogs handelt, ist die Power-On-Flag nicht gesetzt und das Verfahren verzweigt von Abfrageblock 2 zu Abfrageblock 9, wo überprüft wird, ob in dem zweiten Speicherelement (Speicher2) das erste Bitmuster (BITMUSTER1) abgelegt ist.

[0045] Da in dem Funktionsblock 5 in dem zweiten Speicherelement das erste Bitmuster abgelegt wurde, wird nun von dem Abfrageblock 9 zu Funktionsblock 10 verzweigt, wo ein zweites Bitmuster (BITMUSTER2) in dem zweiten Speicherelement (Speicher2) abgelegt wird, wobei sich das zweites Bitmuster von dem ersten Bitmuster unterscheidet.

[0046] Anschließend wird in einem Funktionsblock 11 eine Watchdog-Zeit aus dem Produkt eines Korrekturfaktors K und dem Inhalt des ersten Speicherelements (Speicher1) ermittelt. Die Watchdog-Zeit ist der zeitliche Abstand, der zwischen zwei unmittelbar aufeinanderfolgenden Bediensignalen des Mikrocontrollers für den Watchdog liegt. Der Korrekturfaktor K ist kleiner als 1, vorzugsweise liegt er im Bereich zwischen 0,75 und 0,9. Durch den Korrekturfaktor K werden Toleranzen in dem Mikrocontroller oder in dem Watchdog berücksichtigt, indem die Bedienung des Watchdog zeitlich nach vorne verlagert wird. Dadurch kann ein fälschlicherweise ausgelöstes Rücksetzen des Rechners auf Grund einer zu späten Bedienung des Watchdogs verhindert werden.

[0047] In einem anschließenden Funktionsblock 12 wird eine Watchdog-Bedienung initialisiert mit einer Periode, die der Watchdog-Zeit entspricht. Schließlich wird zu der Bootloaderfunktionalität in Funktionsblock 13 verzweigt.

[0048] Dort wird mit Hilfe des Bootloaders ein Applikationsprogramm in den Programmspeicher des Mikrocontrollers geladen. Mit Hilfe verschiedener Applikationsprogramme kann eine Mikrocontroller-Bauart für verschiedene Steuerungsanwendungen eingesetzt werden. Mit der in Funktionsblock 12 bestimmten Periode wird während der Programmierung des Mikrocontrollers der Watchdog bedient, um ein Zurücksetzen des Mikrocontrollers während der Programmierung zu verhindern. Wenn kein Applikationsprogramm in den Programmspeicher des Mikrocontrollers geladen werden muss, z. B. in einem späteren Einsatz des Steuergerätes, weil das Applikationsprogramm bereits geladen ist, wird von dem Bootloader 13 direkt zu dem Applikationsprogramm gesprungen.

[0049] Nachfolgend wird ein Sonderfall des erfindungsgemäßen Verfahrens beschrieben, bei dem während der Endlosschleife 6 ein Rücksätzen des Mikrocontrollers auf Grund einer Fehlfunktion des Mikrocontrollers veranlasst wird. Die Endlosschleife 6 wird also früher als vorgesehen verlassen, was dazu führt, dass in dem ersten Speicherele-

ment (Speicher1) ein falscher Wert abgelegt ist. Das wiederum führt dazu, dass nach dem erneuten Hochfahren des Rechners in dem Funktionsblock 11 eine falsche Watchdog-Zeit und in dem Funktionsblock 12 eine falsche Periode für die Watchdog-Bedienung festgelegt wird. Während der Programmierung des Mikrocontrollers in dem Funktionsblock 13 kommt es infolgedessen zu einem Rücksetzen des Rechners, da der Watchdog nach einem falschen zeitlichen Muster bedient worden ist. Das Rücksetzen ist in Fig. 2 mit einer gestrichelten Linie 14 dargestellt.

[0050] Nach dem Rücksetzen des Rechners fährt der Rechner erneut hoch. Der Funktionsblock 1 und der Abfrageblock 2 werden wie oben beschrieben durchlaufen. Von dem Abfrageblock 9 aus wird jedoch nicht – wie oben beschrieben – zu dem Funktionsblock 10, sondern zu dem Funktionsblock 3 verzweigt, da in dem vorherigen Durchlauf des Verfahrens in Funktionsblock 10 das zweite Bitmuster (BITMUSTER2) in dem zweiten Speicherelement (Speicher2) abgelegt worden ist. In den Funktionsblöcken 4 bis 7 wird dann wieder die Zeitdauer ermittelt und in dem ersten Speicher (Speicher1) abgelegt, die vom Hochfahren des Rechners bis zu einem Rücksetzen des Rechners auf Grund einer ausbleibenden Bedienung des Watchdogs vergeht. Anschließend wird dann in dem Funktionsblock 11 die richtige Watchdog-Zeit und in dem Funktionsblock 12 die richtige Periode für die Watchdogbedienung berechnet, so dass der Mikrocontroller während der Programmierung des Programmspeichers des Mikrocontrollers mit Hilfe des Bootloaders in Funktionsblock 13 mit dem von dem Watchdog vorgegebenen zeitlichen Muster bedient werden kann.

Patentansprüche

1. Verfahren zum Betreiben mindestens eines Rechners und mindestens einer dem mindestens einen Rechner zugeordneten und nach einem bestimmten zeitlichen Muster von dem Rechner zu bedienenden Überwachungsschaltung (Watchdog), insbesondere für eine Steuer- oder Sicherheitseinrichtung für ein Kraftfahrzeug, wobei der Rechner rückgesetzt wird, falls die Überwachungsschaltung nicht nach dem zeitlichen Muster bedient wird, dadurch gekennzeichnet, dass nach einem Hochfahren (1) des Rechners aus einem Rücksetzmodus in einen Arbeitsmodus eine Zeitbasis ermittelt wird, die nach dem Hochfahren des Rechners bis zu einem Rücksetzen des Rechners aufgrund einer fehlenden Bedienung der Überwachungsschaltung nach dem zeitlichen Muster vergeht, und dass die Überwachungsschaltung in zeitlichen Abständen bedient wird, die in Abhängigkeit von der Zeitbasis ermittelt werden (11).
2. Verfahren nach Anspruch 1, dadurch gekennzeichnet, dass das Verfahren ausgeführt wird bevor mit Hilfe eines in dem Rechner gespeicherten Hilfsprogramme (Bootloader) ein Applikationsprogramm in einen Programmspeicher des Rechners geladen wird (13).
3. Verfahren nach Anspruch 1 oder 2, dadurch gekennzeichnet, dass die zeitlichen Abstände durch Multiplikation der Zeitbasis mit einem Korrekturfaktor (K) ermittelt werden (11).
4. Verfahren nach Anspruch 3, dadurch gekennzeichnet, dass der Korrekturfaktor (K) kleiner als 1, vorzugsweise zwischen 0,75 und 0,9, gewählt wird.
5. Verfahren nach einem der Ansprüche 1 bis 4, dadurch gekennzeichnet, dass zum Ermitteln der Zeitbasis nach dem Hochfahren (1) des Rechners aus einem Rücksetzmodus in einen Arbeitsmodus ein Zähler hochgezählt und der Zählerstand in einem ersten Spei-

cherelement (Speichert), vorzugsweise einem Random-Access-Memory (RAM), abgelegt wird (7).

6. Verfahren nach Anspruch 5, dadurch gekennzeichnet, dass der Zählerstand nach einem erneuten Hochfahren des Rechners nach dem Rücksetzen des Rechners aufgrund einer fehlenden Bedienung der Überwachungsschaltung der Zählerstand aus dem ersten Speicherelement (Speicher) ausgelesen und als Zeitbasis zum Bedienen der Überwachungsschaltung herangezogen wird (11).

7. Verfahren nach Anspruch 6, dadurch gekennzeichnet, dass nach einem Hochfahren (1) des Rechners aus einem Rücksetzmodus in einen Arbeitsmodus ein zweites Speicherelement (Speicher2) mit einem ersten vorgebbaren Bitmuster (BITMUSTER1) belegt wird (5) und der Inhalt des zweiten Speicherelements (Speicher2) zum Erkennen eines erneuten Hochfahrens des Rechners bei jedem Hochfahren (1) des Rechners abgefragt wird (9).

8. Verfahren nach Anspruch 7, dadurch gekennzeichnet, dass nach einem erneuten Hochfahren des Rechners das zweite Speicherelement (Speicher2) mit einer von dem ersten Bitmuster (BITMUSTER1) unterschiedlichen zweiten vorgebbaren Bitmuster (BITMUSTER2) belegt wird (10).

9. Verfahren nach einem der Ansprüche 6 bis 8, dadurch gekennzeichnet, dass bei jedem Hochfahren (1) des Rechners überprüft wird (2), ob eine Markierung (Power-On-Flag) des Rechners zum Erkennen eines Hochfahrens des Rechners aus einem Rücksetzmodus in einen Arbeitsmodus nach einem Einschalten des Rechners gesetzt ist.

10. Steuerelement, insbesondere Read-Only-Memory oder Flash-Memory, für ein Steuergerät insbesondere einer Steuer- oder Sicherheitseinrichtung für ein Kraftfahrzeug, auf dem ein Programm abgespeichert ist, das auf einem Rechner, insbesondere auf einem Mikroprozessor, ablauffähig und zur Ausführung eines Verfahrens nach einem der Ansprüche 1 bis 9 geeignet ist.

11. Steuergerät mit mindestens einem Rechner und mindestens einer dem mindestens einen Rechner zugeordneten und nach einem bestimmten zeitlichen Muster von dem Rechner zu bedienenden Überwachungsschaltung (Watchdog), insbesondere für eine Steuer- oder Sicherheitseinrichtung für ein Kraftfahrzeug, dadurch gekennzeichnet, dass das Steuergerät Mittel zum Ermitteln einer Zeitbasis aufweist, die nach einem Hochfahren des Rechners aus einem Rücksetzmodus in einen Arbeitsmodus bis zu einem Rücksetzen des Rechners aufgrund einer fehlenden Bedienung der Überwachungsschaltung nach dem zeitlichen Muster vergeht, und dass das Steuergerät zweite Mittel zum Bedienen der Überwachungsschaltung in der Zeitbasis entsprechenden zeitlichen Abständen aufweist.

Hierzu 2 Seite(n) Zeichnungen

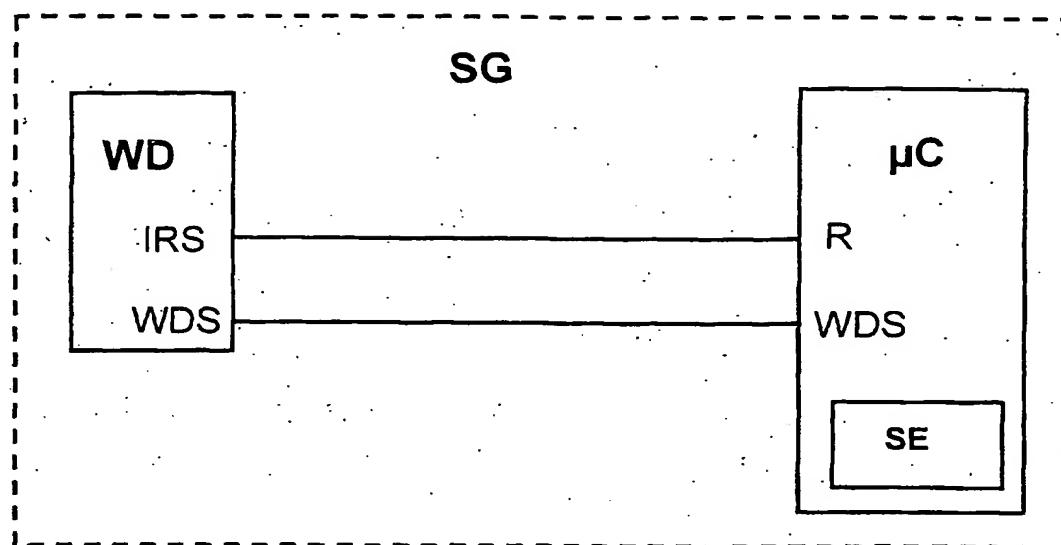
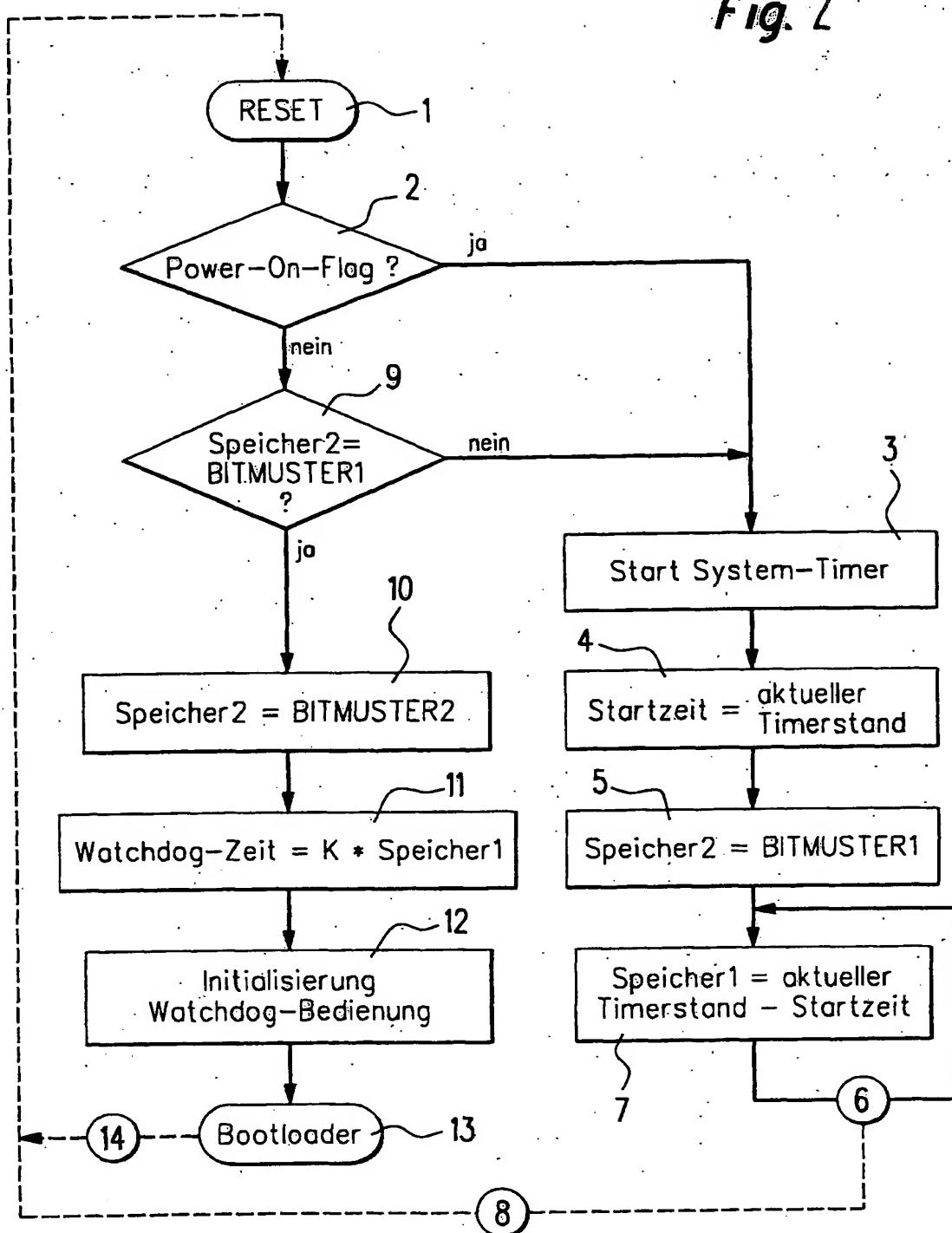


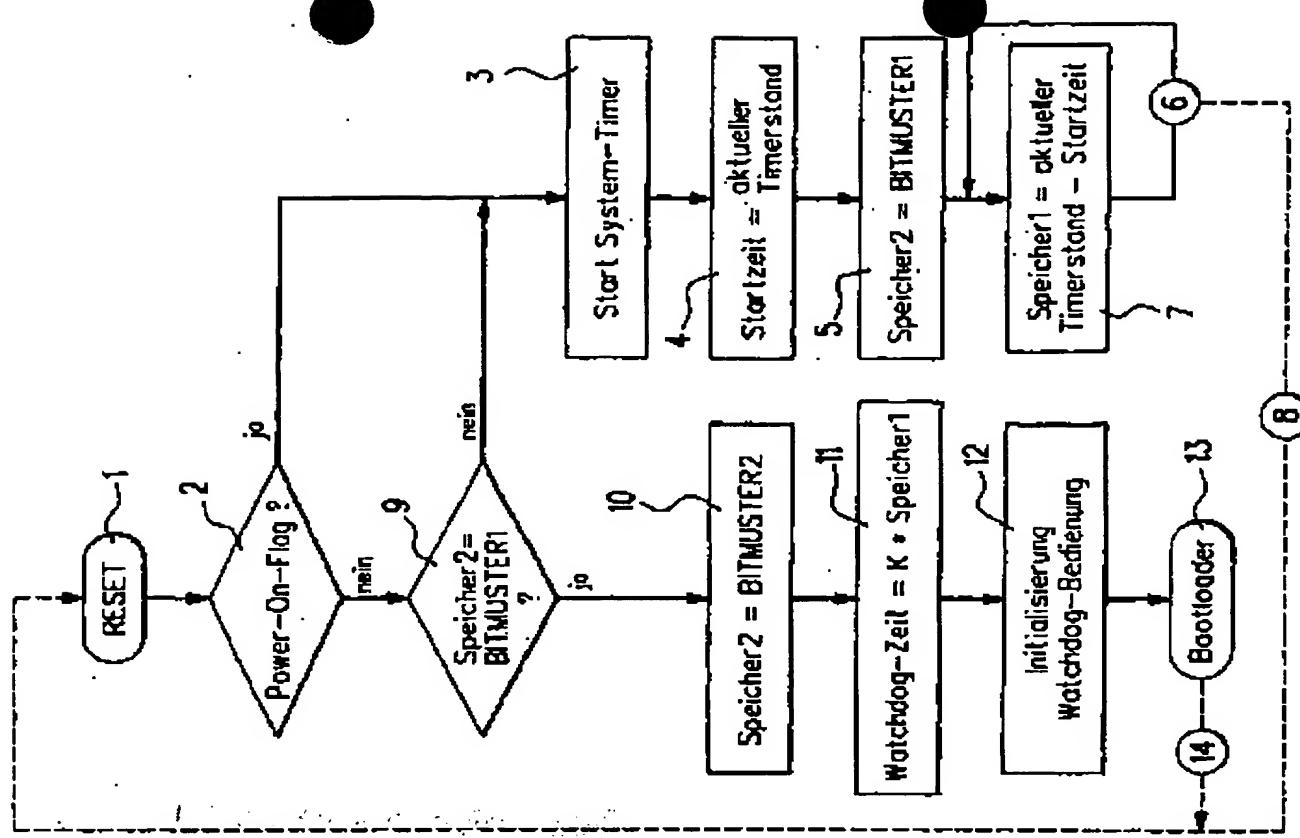
Fig. 1

Fig. 2



AN: PAT 2002-28203
TI: Microcontroller and watchdog operation synchronization method for vehicle control device, involves operating watchdog based on time period elapsed after booting up to resetting operation of microcontroller
PN: DE10030991-A1
PD: 10.01.2002
AB: NOVELTY - The microcontroller is reset if the watchdog is not operated according to a temporal pattern of the microcontroller. The time period elapsed after booting up the microcontroller up to a resetting operation is determined. The watchdog is operated based on the determined time period.
DETAILED DESCRIPTION - INDEPENDENT CLAIMS are also included for the following: (a) Storage device comprising program for synchronizing operation of microcontroller and watchdog; (b) Control device; USE - For synchronizing operation of microcontroller and watchdog used in electronic control unit of vehicle for controlling accelerator, anti lock brake system (ABS) and passenger retaining system such as airbag, return impact damping head support, etc. ADVANTAGE - Prevents resetting of microcontroller during programming by operating watchdog based on determined time period. DESCRIPTION OF DRAWING(S) - The figure shows the flowchart explaining synchronization procedure of operation of watchdog and microcontroller. (Drawing includes non-English language text).
PA: (BOLZ/) BOLZ P; (BOSC) BOSCH GMBH ROBERT;
IN: BOLZ P;
FA: DE10030991-A1 10.01.2002; US6843064 B2 25.01.2005;
US2002046362-A1 18.04.2002;
CO: DE; US;
IC: G05B-023/02; G06F-011/14; G06F-011/30;
MC: T01-F05B2; T01-J07D1; T01-S03; X22-A03B1A; X22-C02C3;
X22-J03A3; X22-J07;
DC: T01; X22;
FN: 2002282038.gif
PR: DE1030991 30.06.2000;
FP: 10.01.2002
UP: 02.02.2005

THIS PAGE BLANK (USPTO)



BEST AVAILABLE COPY

THIS PAGE BLANK (USPTO)

DOCKET NO: 53-02914830
SERIAL NO: 10/535,126
APPLICANT: Grafhoff et al.
LERNER AND GREENBERG P.A.
P.O. BOX 2480
HOLLYWOOD, FLORIDA 33022
TELE: (954) 925-1100